

Multi-product expansion for Nonlinear Differential Equations

Jürgen Geiser

juergen.geiser@uni-greifswald.de

Abstract. In this paper we discuss the extension of MPE methods to nonlinear differential equations. We concentrate on nonlinear systems of differential equations and generalize the recent MPE method, see [16].

Keyword Suzuki's method, Multi-product expansion, exponential splitting, non-autonomous systems, nonlinear differential equations.

AMS subject classifications. 65M15, 65L05, 65M71.

1 Introduction

In this paper we concentrate on approximation to the solution of the nonlinear evolution equation, e.g. nonlinear evolution equation,

$$\partial_t u = F(u(t)) = A(u(t)) + B(u(t)), \quad u(0) = u_0, \quad (1)$$

with the unbounded operators $A : D(A) \subset \mathbf{X} \rightarrow \mathbf{X}$ and $B : D(B) \subset \mathbf{X} \rightarrow \mathbf{X}$. We have further $F(v) = A(v) + B(v)$, $v \in D(A) \cap D(B)$.

We assume to have suitable chosen subspaces of the underlying Banach space $(X, \|\cdot\|_X)$ such that $D(F) = D(A) \cap D(B) \neq \emptyset$

For such equations, we concentrate on comparing the higher order methods to Suzuki's schemes. Here the Suzuki's methods apply factorized symplectic algorithms with forward derivatives, see [11], [12].

The exact solution of the evolution problem (1) is given as:

$$u(t) = \mathcal{E}_F(t, u(0)), \quad 0 \leq t \leq T, \quad (2)$$

with the evolution operator \mathcal{E}_F depending on the actual time t and the initial value $u(0)$.

We apply a formal notation given as:

$$u(t) = \exp(tD_F)u(0), \quad 0 \leq t \leq T, \quad (3)$$

Here the evolution operator $\exp(tD_F)$ and the Lie-derivative D_F associated with F are given as:

$$\exp(tD_F)Gv = G(\mathcal{E}_F(t, v)), \quad 0 \leq t \leq T, \quad D_F Gv = G'(V)F(v) \quad (4)$$

for any unbounded nonlinear operator $G : D(G) \subset X \rightarrow X$ with Frechet derivative G' .

2 Nonlinear Exponential operator splitting methods

In the course of devising numerical algorithms for solving the prototype nonlinear differential equations

$$\partial_t u = A(u) + B(u), \quad u(0) = u_0, \quad (5)$$

where A and B are non-commuting operators,

Strang[38] proposed two second-order algorithms corresponding to approximating

$$\mathcal{T}(\Delta t) = e^{\Delta t(D_A + D_B)} \quad (6)$$

either as

$$S(\Delta t) = \frac{1}{2} (e^{\Delta t D_A} e^{\Delta t D_B} + e^{\Delta t D_B} e^{\Delta t D_A}) \quad (7)$$

or as

$$S_{AB}(\Delta t) = e^{(\Delta t/2)D_B} e^{\Delta t D_A} e^{(\Delta t/2)D_B}. \quad (8)$$

Following up on Strang's work, Burstein and Mirin[8] suggested that Strang's approximations can be generalized to higher orders in the form of a multi-product expansion (MPE),

$$e^{\Delta t(D_A + D_B)} = \sum_k c_k \prod_i e^{a_{ki} \Delta t D_A} e^{b_{ki} \Delta t D_B} \quad (9)$$

and gave two third-order approximations

$$D(\Delta t) = \frac{4}{3} \left(\frac{S_{AB}(\Delta t) + S_{BA}(\Delta t)}{2} \right) - \frac{1}{3} S(\Delta t) \quad (10)$$

where S_{BA} is just S_{AB} with $A \leftrightarrow B$ interchanged, and

$$B_{AB}(\Delta t) = \frac{9}{8} e^{(\Delta t/3)D_A} e^{(2\Delta t/3)D_B} e^{(2\Delta t/3)D_A} e^{(\Delta t/3)D_B} - \frac{1}{8} e^{\Delta t D_A} e^{\Delta t D_B}. \quad (11)$$

They credited J. Dunn for finding the decomposition $D(\Delta t)$ and noted that the weights c_k are no longer positive beyond second order. Thus the stability of the entire algorithm can no longer be inferred from the stability of each component product.

3 Nonlinear Multi-product decomposition

The multi-product decomposition (9) is obviously more complicated than the single product splitting.

By the way after Burstein and Mirin, Sheng[37] proved their observation that beyond second-order, a_{ki} , b_{ki} and c_k cannot all be positive. Some possibilities are the idea of complex coefficients to obtain a higher order scheme.

For general applications, including solving time-irreversible problems, one must have a_{ki} and b_{ki} positive.

Here we show, that we obtain a extrapolation scheme, that can overcome such problems.

Therefore every single product in (9) can at most be second-order[37,39]. But such a product is easy to construct, because every left-right symmetric single product *is* second-order. Let $\mathcal{T}_S(h)$ be such a product with $\sum_i a_{ki} = 1$ and $\sum_i b_{ki} = 1$, then $\mathcal{T}_S(h)$ is time-symmetric by construction,

$$\mathcal{T}_S(-h)\mathcal{T}_S(h) = 1, \quad (12)$$

implying that it has only odd powers of h

$$\mathcal{T}_S(\Delta t) = \exp(\Delta t(D_A + D_B) + \Delta t^3 E_3 + \Delta t^5 E_5 + \dots) \quad (13)$$

and therefore correct to second-order. (The error terms E_i are nested commutators of D_A and D_B depending on the specific form of \mathcal{T}_S .) This immediately suggests that the k th power of \mathcal{T}_S at step size h/k must have the form

$$\mathcal{T}_S^k(\Delta t/k) = \exp(\Delta t(D_A + D_B) + k^{-2}\Delta t^3 E_3 + k^{-4}\Delta t^5 E_5 + \dots), \quad (14)$$

and can serve as a basis for the multi-production expansion (9). The simplest such symmetric product is

$$\mathcal{T}_2(h) = S_{AB}(h) \quad \text{or} \quad \mathcal{T}_2(h) = S_{BA}(h). \quad (15)$$

If one naively assumes that

$$\mathcal{T}_2(h) = e^{\Delta t(D_A + D_B)} + Ch^3 + Dh^4 + \dots, \quad (16)$$

then a Richardson extrapolation would only give

$$\frac{1}{k^2 - 1} [k^2 \mathcal{T}_2^k(h/k) - \mathcal{T}_2(h)] = e^{\Delta t(D_A + D_B)} + O(h^4), \quad (17)$$

a third-order[36] algorithm. However, because the error structure of $\mathcal{T}_2(h/k)$ is actually given by (14), one has

$$\mathcal{T}_2^k(h/k) = e^{\Delta t(D_A + D_B)} + k^{-2}h^3 E_3 + \frac{1}{2}k^{-2}h^4 [(D_A + D_B)E_3 + E_3(D_A + D_B)] + O(h^5), \quad (18)$$

and *both* the third and fourth order errors can be eliminated simultaneously, yielding a fourth-order algorithm. Similarly, the leading $2n+1$ and $2n+2$ order errors are multiplied by k^{-2n} and can be eliminated at the same time. Thus for a given set of n whole numbers $\{k_i\}$ one can have a $2n$ th-order approximation

$$e^{\Delta t(D_A + D_B)} = \sum_{i=1}^n c_i \mathcal{T}_2^{k_i} \left(\frac{\Delta t}{k_i} \right) + O(h^{2n+1}). \quad (19)$$

provided that c_i satisfy the simple Vandermonde equation:

$$\begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ k_1^{-2} & k_2^{-2} & k_3^{-2} & \dots & k_n^{-2} \\ k_1^{-4} & k_2^{-4} & k_3^{-4} & \dots & k_n^{-4} \\ \dots & \dots & \dots & \dots & \dots \\ k_1^{-2(n-1)} & k_2^{-2(n-1)} & k_3^{-2(n-1)} & \dots & k_n^{-2(n-1)} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \dots \\ c_n \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \dots \\ 0 \end{pmatrix} \quad (20)$$

Surprising, this equation has closed form solutions[13] for all n

$$c_i = \prod_{j=1(\neq i)}^n \frac{k_i^2}{k_i^2 - k_j^2}. \quad (21)$$

The natural sequence $\{k_i\} = \{1, 2, 3 \dots n\}$ produces a $2n$ th-order algorithm with the minimum $n(n+1)/2$ evaluations of $\mathcal{T}_2(h)$. For orders four to ten, one has explicitly:

$$\mathcal{T}_4(\Delta t) = -\frac{1}{3}\mathcal{T}_2(\Delta t) + \frac{4}{3}\mathcal{T}_2^2\left(\frac{\Delta t}{2}\right) \quad (22)$$

$$\mathcal{T}_6(\Delta t) = \frac{1}{24}\mathcal{T}_2(\Delta t) - \frac{16}{15}\mathcal{T}_2^2\left(\frac{\Delta t}{2}\right) + \frac{81}{40}\mathcal{T}_2^3\left(\frac{\Delta t}{3}\right) \quad (23)$$

$$\mathcal{T}_8(\Delta t) = -\frac{1}{360}\mathcal{T}_2(\Delta t) + \frac{16}{45}\mathcal{T}_2^2\left(\frac{\Delta t}{2}\right) - \frac{729}{280}\mathcal{T}_2^3\left(\frac{\Delta t}{3}\right) + \frac{1024}{315}\mathcal{T}_2^4\left(\frac{\Delta t}{4}\right) \quad (24)$$

$$\begin{aligned} \mathcal{T}_{10}(\Delta t) = & \frac{1}{8640}\mathcal{T}_2(\Delta t) - \frac{64}{945}\mathcal{T}_2^2\left(\frac{\Delta t}{2}\right) + \frac{6561}{4480}\mathcal{T}_2^3\left(\frac{\Delta t}{3}\right) \\ & - \frac{16384}{2835}\mathcal{T}_2^4\left(\frac{\Delta t}{4}\right) + \frac{390625}{72576}\mathcal{T}_2^5\left(\frac{\Delta t}{5}\right) \dots \end{aligned} \quad (25)$$

As shown in Ref.[13], $\mathcal{T}_4(h)$ reproduces Nyström's fourth-order algorithm with three force-evaluations and $\mathcal{T}_6(h)$ yielded a new sixth-order Nyström type algorithm with five force-evaluations.

3.1 Error Analysis of the Nonlinear Multiproduct Expansion

The error analysis is based on the following ideas:

- Definition of a new operator (linear operator)
- Definition of the derivatives
- Derivation of the splitting errors

The nonlinear equation is given as:

$$\partial_t u = F(u(t)) = A(u(t)) + B(u(t)), \quad u(0) = u_0, \quad (26)$$

We associate a new operator \hat{F} which is a linear Lie operator:

Definition 1. For the given operator F we associate a new operator, denoted \hat{F} .

This Lie-operator acts on the space of the differentiable operators of the type: $S \rightarrow S$,

and maps each operator G into the new operator $\hat{F}(G)$, such that for any element $c \in S$:

$$(\hat{F}(G))(c) = (G'(c) \circ F)(c), \quad (27)$$

Here the derivatives are given as:

Definition 2. *The k -th power of the Lie-operator \hat{F} applied to some operator G can be expressed as the k -th derivative of G , that is the relation*

$$(\hat{F}^k(G))(c) = \frac{\partial^k G(c)}{\partial t^k}, \quad (28)$$

is valid for all $k = 1, 2, \dots$

Let us split the operator F into the sum $F_1 + F_2$
The splitting error for the A-B splitting is given as:

$$Err_{A-B} = \exp(\tau\hat{F}_1 + \hat{F}_2)(I) - (\exp(\tau\hat{F}_1)\exp(\tau\hat{F}_2))(I)(c), \quad (29)$$

if we set in the commutator we obtain for F_1 and F_2

$$Err_{A-B,1,2} = \tau(F'_2(c) \circ F_1)(c) - F'_1(c) \circ F_2(c) + O(\tau^2), \quad (30)$$

The Strang Splitting is given as:

$$\begin{aligned} Err_{Strang} &= \exp(\tau\hat{F}_1 + \hat{F}_2)(I) \\ &\quad - (\exp(\tau/2\hat{F}_1)\exp(\tau\hat{F}_2)\exp(\tau/2\hat{F}_1))(I)(c), \end{aligned} \quad (31)$$

if we set in the commutator we obtain for F_1 and F_2

$$Err_{Strang,1,2} = \frac{1}{24}\tau^2([F_2, [F_2, F_1]](c) - 2[F_1, [F_1, F_2]](c)) + O(\tau^4). \quad (32)$$

with the commutator: $[F_1, F_2](c) = (F'_2(c) \circ F_1)(c) - (F'_1(c) \circ F_2)(c)$.

We derive the nonlinear MPE based on the definition of the linearized Lie-operator.

The derivation of the MPE method is given as:

$$\mathcal{T}_2^k(h/k) = e^{\Delta t(D_A + D_B)} + k^{-2}h^3E_3 + \frac{1}{2}k^{-2}h^4[(D_A + D_B)E_3 + E_3(D_A + D_B)] + O(h^5), \quad (33)$$

and *both* the third and fourth order errors can be eliminated simultaneously, while E_3 and E_4 are given as commutators of the derivatives of A and B , see [16].

4 Nonlinear Operator splitting methods

In the literature, there are various types of splitting methods. We mainly consider the following operators splitting schemes in this study:

1. Sequential operator splitting: A-B splitting

$$\frac{\partial c^*(t)}{\partial t} = A(c^*(t)) \quad \text{with } t \in [t^n, t^{n+1}] \quad \text{and} \quad c^*(t^n) = c_{sp}^n \quad (34)$$

$$\frac{\partial c^{**}(t)}{\partial t} = B(c^{**}(t)) \quad \text{with } t \in [t^n, t^{n+1}] \quad \text{and} \quad c^{**}(t^n) = c^*(t^{n+1}), \quad (35)$$

for $n = 0, 1, \dots, N - 1$ whereby $c_{sp}^n = c_0$ is given from (??). The approximated split solution at the point $t = t^{n+1}$ is defined as $c_{sp}^{n+1} = c^{**}(t^{n+1})$.

2. Strang-Marchuk operator splitting : A-B-A splitting

$$\frac{\partial c^*(t)}{\partial t} = A(c^*(t)) \quad \text{with } t \in [t^n, t^{n+1/2}] \quad \text{and} \quad c^*(t^n) = c_{sp}^n \quad (36)$$

$$\frac{\partial c^{**}(t)}{\partial t} = B(c^{**}(t)) \quad \text{with } t \in [t^n, t^{n+1/2}] \quad \text{and} \quad c^{**}(t^n) = c^*(t^{n+1/2}), \quad (37)$$

$$\frac{\partial c^{***}(t)}{\partial t} = A(c^*(t)) \quad \text{with } t \in [t^{n+1/2}, t^{n+1}] \quad \text{and} \quad c^{***}(t^{n+1/2}) = c^{**}(t^{n+1/2}) \quad (38)$$

where $t^{n+1/2} = t^n + \tau/2$, τ is the local time step. The approximated split solution at the point $t = t^{n+1}$ is defined as $c_{sp}^{n+1} = c^{***}(t^{n+1})$.

3. Iterative splitting with respect to one operator

$$\frac{\partial c_i(t)}{\partial t} = A(c_i(t)) + B(c_{i-1}(t)), \quad \text{with } c_i(t^n) = c^n, i = 1, 2, \dots, m \quad (39)$$

4. Iterative splitting with respect to alternating operators

$$\frac{\partial c_i(t)}{\partial t} = A(c_i(t)) + B(c_{i-1}(t)), \quad \text{with } c_i(t^n) = c^n, \quad (40)$$

$$i = 1, 2, \dots, j,$$

$$\frac{\partial c_i(t)}{\partial t} = A(c_{i-1}(t)) + B(c_i(t)), \quad \text{with } c_{i+1}(t^n) = c^n, \quad (41)$$

$$i = j + 1, j + 2, \dots, m,$$

Here, $c_0(t^n) = c^n$, $c_{-1} = 0$ and c^n is the known split approximation at the time level $t = t^n$. The split approximation at the time-level $t = t^{n+1}$ is defined as $c^{n+1} = c_{2m+1}(t^{n+1})$.

4.1 Error Analysis of the Classical Splitting Schemes

We associate a new operator \hat{F} which is a linear Lie operator:

Definition 3. For the given operator F we associate a new operator, denoted \hat{F} .

This Lie-operator acts on the space of the differentiable operators of the type:
 $S \rightarrow S$,

and maps each operator G into the new operator $\hat{F}(G)$, such that for any element $c \in S$:

$$(\hat{F}(G))(c) = (G'(c) \circ F)(c), \quad (42)$$

Here the derivatives are given as:

Definition 4. The k -th power of the Lie-operator \hat{F} applied to some operator G can be expressed as the k -th derivative of G , that is the relation

$$(\hat{F}^k(G))(c) = \frac{\partial^k G(c)}{\partial t^k}, \quad (43)$$

is valid for all $k = 1, 2, \dots$

Let us split the operator F into the sum $F_1 + F_2$
 The splitting error for the A-B splitting is given as:

$$Err_{A-B} = \exp(\tau \hat{F}_1 + \hat{F}_2)(I) - (\exp(\tau \hat{F}_1) \exp(\tau \hat{F}_2))(I)(c), \quad (44)$$

if we set in the commutator we obtain for F_1 and F_2

$$Err_{A-B,1,2} = \tau(F_2'(c) \circ F_1)(c) - F_1'(c) \circ F_2(c) + O(\tau^2), \quad (45)$$

The Strang Splitting is given as:

$$\begin{aligned} Err_{Strang} &= \exp(\tau \hat{F}_1 + \hat{F}_2)(I) \\ &\quad - (\exp(\tau/2 \hat{F}_1) \exp(\tau \hat{F}_2) \exp(\tau/2 \hat{F}_1))(I)(c), \end{aligned} \quad (46)$$

if we set in the commutator we obtain for F_1 and F_2

$$Err_{Strang,1,2} = \frac{1}{24} \tau^2 ([F_2, [F_2, F_1]](c) - 2[F_1, [F_1, F_2]](c)) + O(\tau^4). \quad (47)$$

with the commutator: $[F_1, F_2](c) = (F_2'(c) \circ F_1)(c) - (F_1'(c) \circ F_2)(c)$.

5 Improving the Initialization of Operator Splitting Methods

A delicate problem in splitting methods is to achieve sufficient accuracy in the first splitting steps, see [24].

Next we discuss the extension to the improvement with Zassenhaus formula.

5.1 Higher order A-B splitting by Initialization

The idea is based on the novel commutator that is given in Section 4.

Theorem 1. *We solve the initial value problem by using the method given in equations (34) and (35). We assume bounded and nonlinear operators A and B .*

The consistency error of the A-B splitting is $\mathcal{O}(t)$, then we can improve the error of the A-B splitting scheme to $\mathcal{O}(t^p)$, $p > 1$ by improving the starting conditions c_0 as

$$c_0 = (\pi_{j=2}^p \exp(\hat{C}_j t^j)) c_0$$

where \hat{C}_j is called a nonlinear Zassenhaus exponents that is given with the novel commutator, e.g. the linear case is given in [?].

The local splitting error of A-B splitting method can be read as follows

$$\begin{aligned} \rho_n &= (\exp(\tau_n(\hat{A} + \hat{B}))(I) - \exp(\tau_n \hat{B})(I) \exp(\tau_n \hat{A})(I))(c_{sp}^n) \\ &= \hat{C}_T \tau_n^{p+1} + \mathcal{O}(\tau_n^{p+2}) \end{aligned} \quad (48)$$

where \hat{C}_T is a function of nonlinear Lie brackets of \hat{A} and \hat{B} .

Proof. Let us consider the subinterval $[0, t]$, where $\tau = t$, the solution of the subproblem (34) is:

$$c^*(t) = \exp(t\hat{A})(I)(c_0) \quad (49)$$

after improving the initialization we have

$$c^*(t) = \exp(t\hat{A})(I)(\pi_{j=2}^p \exp(\hat{C}_j t^j))(I)(c_0) \quad (50)$$

the solution of the subproblem (35) becomes

$$\begin{aligned} c^{**}(t) &= \exp(t\hat{B})(I) \exp(t\hat{A})(I)(\pi_{j=2}^p \exp(\hat{C}_j t^j))(I)c_0 \\ &= (\exp(\tau_n(\hat{B} + \hat{A}))(I)(c_0) + \mathcal{O}(t^{p+1})) \end{aligned} \quad (51)$$

with the help of the Zassenhaus product formula.

Remark 1. For example, the second order A-B splitting after improving the initialization is

$$\begin{aligned} c^{**}(t) &= \left(\exp(t\hat{B})(I) \exp(t\hat{A})(I) \exp(-\frac{1}{2}t^2[\hat{B}, \hat{A}])(I) \right)(c_0) \\ &= \exp(t(\hat{B} + \hat{A}))(I)(c_0) + \mathcal{O}(t^3) \end{aligned} \quad (52)$$

and the third order A-B splitting after improving the initialization is

$$\begin{aligned} c^{**}(t) &= \left(\exp(t\hat{B})(I) \exp(t\hat{A})(I) \exp(-\frac{1}{2}t^2[B, A])(I) \exp(\frac{1}{6}t^3[\hat{B}, [\hat{B}, \hat{A}]](I) \right. \\ &\quad \left. - \frac{1}{3}[\hat{A}, [\hat{A}, \hat{B}]](I)) \right)(c_0) \end{aligned} \quad (53)$$

$$= \exp(t(\hat{B} + \hat{A}))(I)(c_0) + \mathcal{O}(t^4), \quad (54)$$

where the commutator is given as $[A, B](c) = (B'(c) \circ A)(c) - (A'(c) \circ B)(c)$.

Remark 2. The same idea can also be done with the Strang-Splitting method, see the linear case in [24].

6 Alternative Approaches: Iterative Schemes for Linearization

In the following, we discuss the fixed point iteration and Newton's method as alternative approaches to linearize the nonlinear problems.

We solve the nonlinear problem:

$$F(x) = 0, \quad (55)$$

where $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$.

6.1 Fixed-point iteration

The nonlinear equations can be formulated as fixed-point problems:

$$x = K(x), \quad (56)$$

where K is the fixed-point map and is nonlinear, e.g. $K(x) = x - F(x)$.

A solution of (57) is called fix-point of the map K .

The fix-point iteration is given as:

$$x_{i+1} = K(x_i), \quad (57)$$

and is called *nonlinear Richardson iteration*, *Picard iteration*, or *the method of successive substitution*.

Definition 5. Let $\Omega \subseteq \mathbb{R}^n$ and let $G : \Omega \rightarrow \mathbb{R}^m$. G is Lipschitz continuous on Ω with Lipschitz constant γ if

$$\|G(x) - G(y)\| \leq \gamma \|x - y\|, \quad (58)$$

for all $x, y \in \Omega$.

For the convergence we have to assume that K be a contraction map on Ω with Lipschitz constant $\gamma < 1$.

Example 1. We apply the fix-point iterative scheme to decouple the non-separable Hamiltonian problem.

$$\dot{\mathbf{q}}_i = \frac{\partial H}{\partial \mathbf{p}}(\mathbf{p}_i, \mathbf{q}_{i-1}), \quad (59)$$

$$\dot{\mathbf{p}}_i = -\frac{\partial H}{\partial \mathbf{q}}(\mathbf{p}_{i-1}, \mathbf{q}_i), \quad (60)$$

while we have the initial condition for the fix-point iteration:

$$(\mathbf{p}_0, \mathbf{q}_0) = (\mathbf{p}(t^n), \mathbf{q}(t^n))$$

we assume that we have convergent results after $i = 1 \dots, m$ iterative steps

or with the stopping criterion:

$$\max(\|\mathbf{p}_{i+1} - \mathbf{p}_i\|, \|\mathbf{q}_{i+1} - \mathbf{q}_i\|) \leq err,$$

while $\|\cdot\|$ is the Euclidean norm (or a simple vector-norm, e.g. L_2).

6.2 Newton's method

We solve the nonlinear operator equation (55).

While $F : D \subset X \rightarrow Y$ with the Banach spaces X, Y is given with the norms $\|\cdot\|_X$ and $\|\cdot\|_Y$. Let F be at least once continuous differentiable, further we assume x_0 is a starting solution of the unknown solution x^* .

Then the *successive linearization* lead to the general Newton's method:

$$F'(x_i)\Delta x_i = -F(x_i), \quad (61)$$

where $\Delta x_i = x_{i+1} - x_i$ and $i = 0, 1, 2, \dots$.

The method derive the solution of a nonlinear problem by solving *a sequence of linear problems of the same kind*.

Remark 3. The linearization methods can be applied to the MPE methods. Non-separable Hamiltonian problems can decoupled to separable Hamiltonians, see the ideas in [14].

7 Numerical Examples

In the following section, we deal with experiments to verify the benefit of our methods. At the beginning, we propose introductory examples to compare the methods. In the next examples, some ideas to applications in Burgers and Hamiltonian problems, are done.

7.1 First Example: Burgers equation

We deal with a 2D example where we can derive an analytical solution.

$$\partial_t u = -u\partial_x u - u\partial_y u + \mu(\partial_{xx} u + \partial_{yy} u) + f(x, y, t), \quad (62)$$

$$(x, y, t) \in \Omega \times [0, T]$$

$$u(x, y, 0) = u_{\text{ana}}(x, y, 0), \quad (x, y) \in \Omega \quad (63)$$

$$\text{with } u(x, y, t) = u_{\text{ana}}(x, y, t) \text{ on } \partial\Omega \times [0, T], \quad (64)$$

where $\Omega = [0, 1] \times [0, 1]$, $T = 1.25$, and μ is the viscosity.

The analytical solution is given as

$$u_{\text{ana}}(x, y, t) = (1 + \exp(\frac{x + y - t}{2\mu}))^{-1}, \quad (65)$$

where $f(x, y, t) = 0$.

For the non-asymptotic case we compute the right-hand side as:

$$f(x, y, t) = -\partial_t u_{\text{asym}} - u_{\text{asym}}\partial_x u_{\text{asym}} - u_{\text{asym}}\partial_y u_{\text{asym}} \quad (66)$$

$$+ \mu(\partial_{xx} u_{\text{asym}} + \partial_{yy} u_{\text{asym}}) + f(x, y, t), \quad \text{in } (x, y, t) \in [0, 1] \times [0, 1] \times [0, 1.25]$$

$$u(x, y, 0) = u_{\text{asym}}(x, y, 0), \quad (x, y) \in [0, 1] \times [0, 1] \quad (67)$$

$$\text{with } u(x, y, t) = u_{\text{asym}}(x, y, t) \text{ on } \partial\Omega \times [0, T], \quad (68)$$

We discretize with $\Delta x, \Delta y = 1/40$, $\Delta t = 0.001$.
The operators are given as:

$$\begin{aligned} A(u)u &= -u\partial_x u - u\partial_y u, \text{ hence } A(u) = -u\partial_x - u\partial_y \text{ (the nonlinear operator),} \\ Bu &= \mu(\partial_{xx}u + \partial_{yy}u) + f(x, y, t) \text{ (the linear operator).} \end{aligned}$$

We apply the nonlinear Algorithm (40) to the first equation and obtain

$$\begin{aligned} A(u_{i-1})u_i &= -u_{i-1}\partial_x u_i - u_{i-1}\partial_y u_i \text{ and} \\ Bu_{i-1} &= \mu(\partial_{xx} + \partial_{yy})u_{i-1} + f, \end{aligned}$$

and we obtain linear operators, because u_{i-1} is known from the previous time step.

In the second equation we obtain by using Algorithm (41):

$$\begin{aligned} A(u_{i-1})u_i &= -u_{i-1}\partial_x u_i - u_{i-1}\partial_y u_i \text{ and} \\ Bu_{i+1} &= \mu(\partial_{xx} + \partial_{yy})u_{i+1} + f, \end{aligned}$$

and we have also linear operators.

The maximal error at end time $t = T$ is given as

$$\text{err}_{\max} = |u_{\text{num}} - u_{\text{ana}}| = \max_{i=1}^p |u_{\text{num}}(x_i, t) - u_{\text{ana}}(x_i, t)|,$$

the numerical convergence rate is given as

$$\rho = \log(\text{err}_{h/2}/\text{err}_h)/\log(0.5).$$

We have the following results, see Tables 1 for different steps in time and space and different viscosities, see also the results in [23].

7.2 Separable Hamiltonian

We deal with the evolution of any dynamical variable $u(\mathbf{q}, \mathbf{p})$ (including \mathbf{q} and \mathbf{p} themselves) is given by the Poisson bracket,

$$\partial_t u(\mathbf{q}, \mathbf{p}) = \left(\frac{\partial u}{\partial \mathbf{q}} \cdot \frac{\partial H}{\partial \mathbf{p}} - \frac{\partial u}{\partial \mathbf{p}} \cdot \frac{\partial H}{\partial \mathbf{q}} \right) = (A + B)u(\mathbf{q}, \mathbf{p}). \quad (69)$$

An example for a separable Hamiltonian is give as:

$$H(\mathbf{p}, \mathbf{q}) = \frac{\mathbf{p}^2}{2m} + V(\mathbf{q}), \quad (70)$$

A and B are Lie operators, or vector fields

$$A = \mathbf{v} \cdot \frac{\partial}{\partial \mathbf{q}} \quad B = \mathbf{a}(\mathbf{q}) \cdot \frac{\partial}{\partial \mathbf{v}} \quad (71)$$

| $\Delta x = \Delta y$ | Δt | err $_{L_1}$ | err $_{\max}$ | ρ_{L_1} | ρ_{\max} |
|-----------------------|------------|--------------|---------------|--------------|---------------|
| 1/10 | 1/10 | 0.0549 | 0.1867 | | |
| 1/20 | 1/10 | 0.0468 | 0.1599 | 0.2303 | 0.2234 |
| 1/40 | 1/10 | 0.0418 | 0.1431 | 0.1630 | 0.1608 |
| 1/10 | 1/20 | 0.0447 | 0.1626 | | |
| 1/20 | 1/20 | 0.0331 | 0.1215 | 0.4353 | 0.4210 |
| 1/40 | 1/20 | 0.0262 | 0.0943 | 0.3352 | 0.3645 |
| 1/10 | 1/40 | 0.0405 | 0.1551 | | |
| 1/20 | 1/40 | 0.0265 | 0.1040 | 0.6108 | 0.5768 |
| 1/40 | 1/40 | 0.0181 | 0.0695 | 0.5517 | 0.5804 |

Table 1. Numerical results for the Burgers equation with viscosity $\mu = 0.05$, initial condition $u_0(t) = c_n$, and two iterations per time step.

where we have abbreviated $\mathbf{v} = \mathbf{p}/m$ and $\mathbf{a}(\mathbf{q}) = -\nabla V(\mathbf{q})/m$. The exponential operators e^{hA} and e^{hB} are then just shift operators.

$$S(h) = e^{h/2B} e^{hA} e^{h/2B}$$

That is also given as a Verlet-algorithm in the following scheme.
We start with $(\mathbf{q}_0, \mathbf{v}_0)^t = (\mathbf{q}(t^n), \mathbf{v}(t^n))^t$:

$$(\mathbf{q}_1, \mathbf{v}_1)^t = e^{h/2B}(\mathbf{q}_0, \mathbf{v}_0)^t = (I + \frac{1}{2}h \sum_i a(\mathbf{q}) \frac{\partial}{\partial \mathbf{v}_i})(\mathbf{q}_0, \mathbf{v}_0)^t \quad (72)$$

$$= (\mathbf{q}_0, \mathbf{v}_0 + \frac{1}{2}ha(\mathbf{q}_0))^t, \quad (73)$$

$$(\mathbf{q}_2, \mathbf{v}_2)^t = e^{hA}(\mathbf{q}_1, \mathbf{v}_1)^t = (I + h \sum_i \mathbf{v}_i \frac{\partial}{\partial \mathbf{q}_i})(\mathbf{q}_1, \mathbf{v}_1)^t \quad (74)$$

$$= (\mathbf{q}_1 + h\mathbf{v}_1, \mathbf{v}_1)^t, \quad (75)$$

$$(\mathbf{q}_3, \mathbf{v}_3)^t = e^{h/2B}(\mathbf{q}_2, \mathbf{v}_2)^t = (I + \frac{1}{2}h \sum_i a(\mathbf{q}) \frac{\partial}{\partial \mathbf{v}_i})(\mathbf{q}_2, \mathbf{v}_2)^t \quad (76)$$

$$= (\mathbf{q}_2, \mathbf{v}_2 + \frac{1}{2}ha(\mathbf{q}_1))^t. \quad (77)$$

And the substitution is given the algorithm for one time-step $n \rightarrow n+1$:

$$(\mathbf{q}_3, \mathbf{v}_3)^t = (\mathbf{q}_0 + h\mathbf{v}_0 + \frac{h}{2}a(\mathbf{q}_0), \mathbf{v}_0 + \frac{h}{2}a(\mathbf{q}_0) + \frac{h}{2}a(\mathbf{q}_0 + h\mathbf{v}_0 + \frac{h}{2}a(\mathbf{q}_0)))^t \quad (78)$$

while $(\mathbf{q}(t^{n+1}), \mathbf{v}(t^{n+1}))^t = (\mathbf{q}_3, \mathbf{v}_3)^t$.

Remark 4. Here, we linearize with respect to the time-steps and assume to compute a large time sequence.

The numerical error is given with $\mathcal{O}(h^2)$ based on the second order approaches of the Strang-splitting.

8 Conclusions and Discussions

We have presented novel MPE approaches to nonlinear differential equations. Based on the ideas of iterative splitting schemes, we could linearize the numerical scheme and apply the linear MPE approach.. Numerical examples confirm the applications to nonlinear equations. In the future we will focus us on the development of improved MPE methods with respect to non-separable Hamiltonian problems.

References

1. Albrecht, J. (1955) Beiträge zum Runge-Kutta-Verfahren. *Zeitschrift für Angewandte Mathematik und Mechanik*, **35**, 100-110 , reproduced in Ref.[2].
2. Battin, R.H. (1999) *An Introduction to the Mathematics and Methods of Astrodynamics, Revised Edition*, AIAA.
3. Blanes, S., Casas, F. & Ros, J. (1999) Symplectic integration with processing: A general study. *SIAM J. Sci., Comp.*, **21**, 711-727.
4. Blanes, S. & Moan, P.C. (2002) Practical symplectic partition Runge-Kutta methods and Runge-Kutta Nyström methods. *J. Comput. Appl. Math.*, **142**, 313-330.
5. Blanes, S., Casas, F., Oteo, J.A. & J. Ros, J. (2009) The Magnus expansion and some of its applications. *Physics Reports*, **470**(5-6), 151-238.
6. Blanes, S., Casas, F. & Ros, J. (1999) Extrapolation of symplectic integrators, *Celest. Mech. Dyn. Astron.*, **75**, 149-161.
7. Brankin, R.W., Gladwell, I., Dormand, J.R., Prince, P.J. & Seward, W.L. (1989) Algorithm 670: a Runge-Kutta-Nyström code, *ACM Trans. Math. Softw. (TOMS)*, **15**, 31-40 .
8. Burstein, S.Z. & Mirin, A.A. (1970) Third order difference methods for hyperbolic equations. *J. Comp. Phys.*, **5**, 547-571.
9. Chan, R. & Murus, A. (2000) Extrapolation of symplectic methods for Hamiltonian problems. *Appl. Numer. Math.*, **34**, 189-205.
10. Chin, S.A. (1997) Symplectic Integrators From Composite Operator Factorizations. *Phys. Lett.*, **A226**, 344-348.
11. Chin, S.A. & Chen, C.R. (2002) Gradient symplectic algorithms for solving the Schrödinger equation with time-dependent potentials. *Journal of Chemical Physics*, **117**(4), 1409-1415.
12. Chin, S.A. & Anisimov, P. (2006) Gradient Symplectic Algorithms for Solving the Radial Schrödinger Equation. *J. Chem. Phys.*, **124**(5), 054106.
13. Chin, S.A. (2010) Multi-product splitting and Runge-Kutta-Nyström integrators. *Celest. Mech. Dyn. Astron.*, **106**, 391-406.
14. S.A. Chin. *Symplectic and energy-conserving algorithms for solving magnetic field trajectories*. Phys. Review E, vol. 77, 066401, 2008.

15. Chin, S.A., Janecek, S. & Krotscheck, E. (2009) Any order imaginary time propagation method for solving the Schrödinger equation. *Chem. Phys. Lett.*, **470**, 342-346.
16. S. Chin and J. Geiser. *Multi-product operator splitting as a general method of solving autonomous and non-autonomous equations*. IMA J. Numer. Anal., first published online January 12, 2011.
17. Dormand, J., El-Mikkawy, M. & Prince, P. (1987) High-Order embedded Runge-Kutta-Nyström formulae. *IMA J. Numer. Analysis*, **7**, 423-430.
18. Dvoretzky, A. & Rogers, C.A. (1950) Absolute and Unconditional Convergence in Normed Linear Spaces. *Proc. Natl. Acad. Sci. U S A*, **36**(4), 192-197.
19. Dyson, F.J. (1976) The radiation theorem of Tomonaga. *Swinger and Feynman, Phys. Rev.*, **75**, 486-502.
20. Eggleston, H.G. (1953) Some Remarks on Uniform Convergence. *Proceedings of the Edinburgh Mathematical Society (Series 2)*, Cambridge University Press, **10** 43-52.
21. Forest, E. & Ruth, R.D. (1990) Fourth-order symplectic integration. *Physica D*, **43**(1), 105-117.
22. Geiser, J. (2008) Fourth-order splitting methods for time-dependent differential equation. *Numerical Mathematics: Theory, Methods and Applications*. Global science press, Hong Kong, China, **1**(3), 321-339.
23. J. Geiser. *Iterative operator-splitting methods for nonlinear differential equations and applications*. NMPDE, published online, March 2010.
24. J. Geiser, G. Tanoglu and N. Guecueyen. *Higher Order Operator-Splitting Methods via Zassenhaus product formula: Theory and Applications*. Computers and Mathematics with Applications, Elsevier, North Holland, accepted July, 2011.
25. Gragg, W.B. (1965) On extrapolation algorithms for ordinary initial value problems. *SIAM J. Numer. Anal.* **2**, 384-404.
26. Hansen, E. & Ostermann, A. (2009) Exponential splitting for unbounded operators. *Mathematics of Computation*, **78**(267), 1485-1496.
27. Hairer, E., Lubich, C., & Wanner, G. (2002) *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. SCM, Springer-Verlag Berlin-Heidelberg-New York, No. 31.
28. Hairer, E., Norsett, S.P. & Wanner, G. (1993) *Solving Ordinary Differential Equations I - Nonstiff Problems*, Second Edition, Springer-Verlag, Berlin.
29. Hildebrand, F.B. (1987) *Introduction to Numerical Analysis*. Second Edition, Dover Edition.
30. Hochbruck, M. & Lubich, C. (2003) On Magnus Integrators for Time-Dependent Schrödinger Equations. *SIAM Journal on Numerical Analysis*, **41**(3), 945-963.
31. Jahnke, T. & Lubich, C. (2000) Error bounds for exponential operator splittings. *BIT Numerical Mathematics*, **40**(4), 735-745.
32. McLachlan, R.I. (1995) On the numerical integration of ordinary differential equations by symmetric composition methods. *SIAM J. Sci. Comput.*, **16**(1), 151-168.
33. Moan, P.C. & Niesen, J. (2008) Convergence of the Magnus series. *J. Found. of Comp. Math.*, **8**(3), 291-301.
34. Neri, F. (1987) Lie Algebra and Canonical Integration. *Technical Report*, Department of Physics, University of Maryland.
35. Nyström, E.J. (1925) Über die Numerische Integration von Differentialgleichungen. *Acta Societatis Scientiarum Ferrica*, **50**, 1-55.
36. Schatzman, M. (1994) Higher order alternate direction methods. *Compt. Meth. Appl. Mech. Eng.*, **116**, 219-225 .

37. Sheng, Q. (1989) Solving linear partial differential equations by exponential splitting. *IMA Journal of Numer. Analysis*, **9**, 199-212.
38. Strang, G. (1968) On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.*, **5**, 506-517.
39. Suzuki, M. (1991) General theory of fractal path-integrals with applications to many-body theories and statistical physics. *J. Math. Phys.*, **32**, 400.
40. Suzuki, M. (1993) General Decomposition Theory of Ordered Exponentials. *Proc. Japan Acad.*, **69**, Ser. B, 161.
41. Suzuki, M. (1996) New scheme of hybrid Exponential product formulas with applications to quantum Monte Carlo simulations. *Computer Simulation Studies in Condensed Matter Physics VIII*, eds, D. Landau, K. Mon and H. Shuttler, Springer, Berlin, 1-6.
42. Wiebe, N., Berry, D.W., Hoyer, P. & Sanders, B.C. (2008) Higher Order Decompositions of Ordered Operator Exponentials. *arXiv.org:0812.0562*.
43. Yoshida, K. (1980) *Functional Analysis*. Classics in Mathematics, Berlin-Heidelberg-New York: Springer.
44. Yoshida, H. (1990) Construction of higher order symplectic integrators. *Physics Letters A*, **150**(5,6,7), 262-268.
45. Zillich, R.E., Mayrhofer, J.M. & Chin, S.A. (2010) Extrapolated high-order propagator for path integral Monte Carlo simulations. *J. Chem. Phys.*, **132**, 044103.